Amendments to the Claims:

This listing of claims replaces all prior versions and listings of claims in the application:

Listing of Claims:

1. (Currently amended) A computer program product residing on a computer readable storage medium comprising instructions, including a branch instruction that when executed on a computing device causes the computing device to:

cause an instruction stream to branch to another instruction in the instruction stream based on a bit of a ~~specified~~ register being set or cleared, the branch instruction specifying the register and the ~~which~~ bit of the ~~specified~~ register to use as a branch control bit.

2. (Previously presented) The computer program product of claim 1, wherein the branch instruction comprises

a bit_postion field that specifies the bit position of the branch control bit in a longword contained in a register.

3. (Previously presented) The computer program product of claim 1, wherein the branch instruction comprises

a branch target field specified as a label in the instruction.

4. (Currently amended) The computer program product of claim 1, wherein the branch instruction comprises

~~an optional~~ a token that is set by a programmer and specifies a number i of instructions to execute following the branch instruction before performing the branch operation.

5. (Currently amended) The computer program product of claim 1, wherein the branch instruction comprises

~~an optional~~ a token that is set by a programmer and specifies a number i of instructions to execute following the branch instruction before performing the branch operation where the number of instructions can be specified as one, two or three.

6. (Previously presented) The computer program product of claim 1 wherein the register is a context-relative transfer register or a general-purpose register that holds the operand.

7. (Currently amended) The computer program product of claim 1, wherein the branch instruction comprises

~~an optional~~ a token that is set by a programmer and which specifies a guess branch prefetch for the instruction for the "branch taken" condition rather than the next sequential instruction.

8. (Cancelled)

9. (Previously presented) The computer program product of claim 1 wherein the branch instruction allows a programmer to select which bit of the register to use to determine the branch operation.

10. (Previously presented) The computer program product of claim 1 wherein the branch instruction allows branches to occur based on evaluation of a bit that is in a data path of a processor.

11. (Previously presented) A method of operating a processor comprising:

evaluating a bit of a register designated to use as a branch control bit, the bit of the register and the register being specified in a branch instruction; and

performing a branching operation based on the specified bit of the specified register being set or cleared.

12. (Original) The method of claim 11 wherein the specified bit position is in a longword contained in a register.

13. (Previously presented) The method of claim 11 further comprising:

branching to another instruction at a branch target field specified as a label in the instruction.

14. (Original) The method of claim 11 wherein the specified bit is specified by a programmer.

15. (Currently amended) The method of claim 11 further comprising:

executing a number i of instructions following execution of the branch instruction before performing the branch operation based on evaluating ~~an optional~~ a token that is set by a programmer.

16. (Original) The method of claim 11 wherein the register is a context-relative transfer register or a general-purpose register that holds the operand.

17. (Currently amended) The method of claim 11 further comprising:

prefetching a branch taken instruction based on ~~an optional~~ a token that is set by a programmer, and which specifies a guess_branch prefetch for the instruction for the "branch taken" condition rather than the next sequential instruction.

18. (Cancelled)

19. (Original) The method of claim 11 wherein the instruction allows a programmer to select which bit of the specified register to use to determine the branch operation.

20. (Original) The method of claim 11 wherein branch evaluation occurs based on evaluation of bits that are in a data path of the processor.

21. (Currently amended) A processor comprising:

a register stack;

an arithmetic logic unit coupled to the register stack and a program control store that stores a branch instruction that causes the processor to:

evaluate a bit of one of the registers of the register stack, the ~~specified~~ bit designated to use as a branch control bit, the bit and the one of the registers of the stack being specified in the branch instruction; and

perform a branching operation specified by the branch instruction based on the ~~specified~~ bit of the register being set or cleared.

22. (Original) The processor of claim 21 wherein the specified bit is in a longword in a general purpose register.

23. (Previously presented) The processor of claim 21, wherein the branch instruction comprises

a branch target field specified as a label in the branch instruction.

24. (Original) The processor of claim 21 wherein the specified bit is specified by a programmer.

25. (Previously presented) The processor of claim 21 wherein the one of the registers is a context-relative transfer register or a general-purpose register that holds an operand.